CLAIMS

1. A method for storing data entered by a user in a remote relational database, the method comprising the steps of:

saving data as a plurality of software components at a server;

converting the plurality of software components into a first string and a second string wherein the first string comprises a markup language format that substantially mimics the software components and the second string comprises a serialized string format of the plurality of software components;

compressing the first string and the second string;

transmitting the compressed first string and the compressed second string to a receiving server; and

storing the compressed first string and the compressed second string in a relational database.

2. The method of claim 1, wherein the markup language format uses string concatenation.

3. The method of claim 1, wherein the compressed first string and the compressed second string are stored in a document data table.

4. The method of claim 3, wherein the compressed first string and the compressed second string are stored as a single record within the document data table.

5. The method of claim 4, wherein the compressed first string and the compressed second string are in binary format.

6. A method for retrieving data from a remote relational database, as requested by a user at a user location, the method comprising the steps of:

requesting data from a relational database through a requesting server;

retrieving a compressed first string and a compressed second string from a relational database;

transmitting the compressed first string and the compressed second string to the requesting server;

decompressing the compressed first string and the compressed second string;

converting the second string to an original plurality of software components wherein the second string represents a serialized string format of the plurality of software components comprising a string of characters;

determining whether the second string was converted;

converting the first string to an original plurality of software components if the second string was not converted, wherein the first string represents a markup language format that substantially mimics the software components; and

displaying the original plurality of software components via a user interface.

7. The method of claim 6, wherein the markup language format uses string concatenation.

8. The method of claim 6, wherein the compressed first string and the compressed second string are retrieved from a document data table.

9. The method of claim 8, wherein the compressed first string and the compressed second string retrieved as a single record within the document data table.

10. The method of claim 9, wherein the compressed first string and the compressed second string are in binary format.

11. A system for storing data entered by a user in a remote relational database, the system comprising:

a server for saving data as a plurality of software components wherein the data is entered by the user;

a convert module for converting the plurality of software components into a first string and a second string wherein the first string comprises a markup language format that substantially mimics the software components and the second string comprises a serialized string format of the plurality of software components;

a compress module for compressing the first string and the second string;

a transmit module for transmitting the compressed first string and the compressed second string to a receiving server; and

a relational database for storing the compressed first string and the compressed second string.

12. The system of claim 11, wherein the markup language format uses string concatenation.

13. The system of claim 11, wherein the compressed first string and the compressed second string are stored in a document data table.

14. The system of claim 13, wherein the compressed first string and the compressed second string are stored as a single record within the document data table.

15. The system of claim 14, wherein the compressed first string and the compressed second string are in binary format.

16. A system for retrieving data from a remote relational database, as requested by a user at a user location, the system comprising:

a requesting server for requesting data from a relational database;

a retrieve module for retrieving a compressed first string and a compressed second string from a relational database;

a transmit module for transmitting the compressed first string and the compressed second string to the requesting server;

a decompress module for decompressing the compressed first string and the compressed second string;

a convert module for converting the second string to an original plurality of software components wherein the second string represents a serialized string format of the plurality of software components comprising a string of characters; determining whether the second string was converted; and converting the first string to an original plurality of software components if the second string was not converted, wherein the first string represents a markup language format that substantially mimics the software components; and

a user interface for displaying the original plurality of software components.

17. The system of claim 16, wherein the markup language format uses string concatenation.

18. The system of claim 16, wherein the compressed first string and the compressed second string are retrieved from a document data table.

19. The system of claim 18, wherein the compressed first string and the compressed second string retrieved as a single record within the document data table.

20. The system of claim 19, wherein the compressed first string and the compressed second string are in binary format.

21. At least one processor readable carrier for storing a computer program of instructions configured to be readable by at least one processor for instructing the at least one processor to execute a computer process for performing the method as recited in claim 1.

22. At least one processor readable carrier for storing a computer program of instructions configured to be readable by at least one processor for instructing the at least one processor to execute a computer process for performing the method as recited in claim 6.

23. At least one signal embodied in at least one carrier wave for transmitting a computer program of instructions configured to be readable by at least one processor to execute a computer process for storing data entered by a user in a remote relational database, the computer process comprising:

saving means for saving data as a plurality of software components at a server;

converting means for converting the plurality of software components into a first string and a second string wherein the first string comprises a markup language format that substantially mimics the software components and the second string comprises a serialized string format of the plurality of software components;

compressing means for compressing the first string and the second string;

transmitting means for transmitting the compressed first string and the compressed second string to a receiving server; and

storing means for storing the compressed first string and the compressed second string in a relational database.

24. At least one signal embodied in at least one carrier wave for transmitting a computer program of instructions configured to be readable by at least one processor to execute a computer process for retrieving data from a remote relational database, as requested by a user at a user location, the computer process comprising:

requesting means for requesting data from a relational database through a requesting server;

retrieving means for retrieving a compressed first string and a compressed second string from a relational database;

transmitting means for transmitting the compressed first string and the compressed second string to the requesting server;

decompressing means for decompressing the compressed first string and the compressed second string;

converting means for converting the second string to an original plurality of software components wherein the second string represents a serialized string format of the plurality of software components comprising a string of characters;

determining means for determining whether the second string was converted;

converting means for converting the first string to an original plurality of software components if the second string was not converted, wherein the first string represents a markup language format that substantially mimics the software components; and

displaying means for displaying the original plurality of software components via a user interface.